

12
B.S.

ADA017711

GENERALIZED TRAJECTORY SIMULATION

Volume I: Overview

TRAJECTORY ANALYSIS PROGRAMMING DEPARTMENT
Information Processing Division
Engineering Science Operations
The Aerospace Corporation
El Segundo, Calif. 90245

21 November 1975

Final Report

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED

DEC 28 1975
RECEIVED

Prepared for

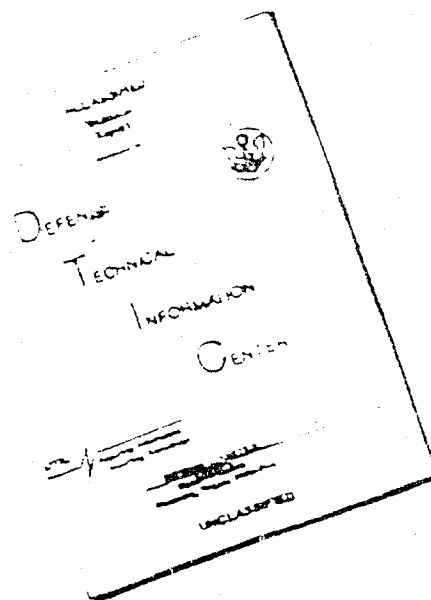
AIR FORCE ROCKET PROPULSION LABORATORY
AIR FORCE SYSTEMS COMMAND
Edwards Air Force Base, Calif. 93523

and THE AEROSPACE CORPORATION

SPACE AND MISSILE SYSTEMS ORGANIZATION
AIR FORCE SYSTEMS COMMAND
Los Angeles Air Force Station
P.O. Box 92960, Worldway Postal Center
Los Angeles, Calif. 90009



DISCLAIMER NOTICE




THIS DOCUMENT IS BEST
QUALITY AVAILABLE. THE COPY
FURNISHED TO DTIC CONTAINED
A SIGNIFICANT NUMBER OF
PAGES WHICH DO NOT
REPRODUCE LEGIBLY.

REPRODUCED FROM
BEST AVAILABLE COPY


This report was prepared by The Aerospace Corporation, El Segundo, California, under Contract F04701-75-C-0076 with the Space and Missile Systems Organization (AFSC), P.O. Box 92960, Worldway Postal Center, Los Angeles, California 90009. The work was sponsored by the Air Force Rocket Propulsion Laboratory (AFSC), Edwards AFB, California 93523, with Capt. Francis Lymburner as project officer. This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public including foreign nationals.


Approved by


A. R. Sims, Director
Mathematics and Programming
Subdivision
Information Processing Division
Engineering Science Operations


This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER


Gerhard E. Aichinger
Technical Advisor
Contracts Management Office


Frank J. Bane
Chief, Contracts Management Office

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM															
1. REPORT NUMBER SAMSOTR-75-255 Vol. I	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER															
4. TITLE (and Subtitle) The Generalized Trajectory Simulation System. Volume I: Overview		5. TYPE OF REPORT & PERIOD COVERED Final Report 1-1-75 to 8-31-75															
6. AUTHOR(s) Trajectory Analysis Programming Department Information Processing Division Engineering Science Operations		7. PERFORMING ORG. REPORT NUMBER TR-0076(6666)-1 Vol. I															
8. AUTHORING ORGANIZATION NAME AND ADDRESS The Aerospace Corporation El Segundo, California 90245		9. CONTRACT OR GRANT NUMBER(s) F04701-74-C-0075, F04701-75-C-0070															
10. CONTROLLING OFFICE NAME AND ADDRESS Space and Missile Systems Organization Air Force Systems Command Los Angeles, Calif. 90045		11. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 156075-31 Aug 75															
12. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Air Force Rocket Propulsion Laboratory Air Force Systems Command Edwards Air Force Base, Calif. 93523		13. REPORT DATE 21 Nov. 1975															
		14. NUMBER OF PAGES 45															
		15. SECURITY CLASS. (of this report) Unclassified															
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE															
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.																	
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)																	
18. SUPPLEMENTARY NOTES																	
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) <table border="0"> <tr> <td>CDC 6000/7000</td> <td>Remote Terminal Access</td> <td>Launch Support</td> </tr> <tr> <td>Modularized Program</td> <td>Trajectory Analysis</td> <td>Error Analysis</td> </tr> <tr> <td>Software System</td> <td>Flight Dynamics</td> <td>Optimization</td> </tr> <tr> <td>Physical Modeling</td> <td>Boost Vehicle Simulation</td> <td>Least Squares</td> </tr> <tr> <td>Verification and Validation</td> <td>Reentry Vehicle Simulation</td> <td>Interpolation</td> </tr> </table>			CDC 6000/7000	Remote Terminal Access	Launch Support	Modularized Program	Trajectory Analysis	Error Analysis	Software System	Flight Dynamics	Optimization	Physical Modeling	Boost Vehicle Simulation	Least Squares	Verification and Validation	Reentry Vehicle Simulation	Interpolation
CDC 6000/7000	Remote Terminal Access	Launch Support															
Modularized Program	Trajectory Analysis	Error Analysis															
Software System	Flight Dynamics	Optimization															
Physical Modeling	Boost Vehicle Simulation	Least Squares															
Verification and Validation	Reentry Vehicle Simulation	Interpolation															
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) <p>The Generalized Trajectory Simulation (GTS) system provides a vehicle design and trajectory simulation capability. GTS is written in FORTRAN compatible with CDC 6000/7000 series computer systems. User-oriented input data specifications, computational efficiency, diverse program applicability, and convenient program modifications have been primary considerations in the design of the GTS system. The trajectory simulation capability can accommodate diverse types of vehicle configurations, flight</p>																	

DD FORM 1473

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

404 068

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

19 KEY WORDS (Continued)

Guidance and Targeting	Post Flight Reconstruction	Vehicle Sizing
Trajectory Simulation	Numerical Operators	Vehicle Design
Flight Mechanics	Numerical Integration	Pre-processors
Powered Flight Simulation	Nonlinear Programming	Symbolic Input
Orbital Analysis	Solid Rocket Motor Sizing	Language
Performance Analysis	Vehicle Weight Estimation	
Dynamic Systems Simulation	Propellant Technology	

20 ABSTRACT (Continued)

profiles, and mission objectives. Additionally, the GTS system contains an extensive vehicle sizing capability and a state-of-the-art optimization capability. The computer memory requirement depends upon the application of the system. GTS applications on 6000 series computers typically require 125K to 160K octal words of memory. For the 7000 series computers, GTS applications typically require 120K to 140K octal words of small core memory (SCM) and 50K to 100K octal words of large core memory (LCM).

This volume provides the potential user an overview of the GTS system. Included is a summary of the major operational capabilities and the structural design of the GTS system.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

PREFACE

This volume, the first of five volumes that describe the Generalized Trajectory Simulation (GTS) system, provides the potential user with an overview of GTS, including a summary of the major operational capabilities and structural design of the GTS system. The remaining volumes are:

Volume II: GTS Usage Guide. This document serves as a general usage guide for GTS and includes a set of example problems, a comprehensive description of the Generalized Trajectory Language, and a discussion of the trajectory simulation control. In addition, a set of appendices contain a master reference list for all volumes and supplementary information to aid the user in defining his problem.

Volume III: GTS Flight Dynamic Models. This publication deals with the GTS library of flight mechanics and flight dynamics models utilized for trajectory simulations.

Volume IV: GTS Numerical Operators. This report concerns the GTS library of numerical operators, including integration, optimization, and interpolation operators.

Volume V: GTS Weight Estimation Models for Sizing Applications. This volume documents the GTS library of weight estimation models utilized for sizing applications.

This report was prepared by D. S. Meder and J. L. Searcy.

CONTENTS

1.	INTRODUCTION	1
1.1	GTS Operational Capabilities	2
1.1.1	Trajectory Simulations.	2
1.1.2	Optimization	5
1.1.3	Vehicle Sizing Analysis	5
1.1.4	Generalized Trajectory Language (GTL)	5
2.	GTS SYSTEM CHARACTERISTICS	7
2.1	Trajectory Applications	7
2.1.1	Model Types and Models.	7
2.1.2	Trajectory Profile Specification.	9
2.2	Sizing Applications	10
2.3	Numerical Operators	11
2.3.1	Integration Techniques.	11
2.3.2	Optimization Capability	12
2.4	General Applicability of the GTS System	13
2.5	Generalized Trajectory Language (GTL)	14
2.6	Input/Output Capabilities	17
2.7	Trajectory Simulation Example	19
2.8	Remote Terminal Execution of GTS.	29
3.	GTS SYSTEM ARCHITECTURE	31
3.1	Control Card Generator	31
3.2	GTL Processor	32
3.3	GTS Pre-Processor.	32
3.4	GTS System Executives and Model Libraries.	32
3.5	GTS Post-Processors	33

FIGURES

1. Orbit Configuration	20
2. GTL Input Data	21
3. Remote Terminal Input	30

TABLES

1. Principal Components of the GTS System	3
2. Representative Trajectory Simulation Applications	4
3. Trajectory Simulation Model Types in GTS	8

SECTION 1

INTRODUCTION

This document is the first of a set of five volumes that describe the Generalized Trajectory Simulation (GTS) system. The intent of the first volume is to provide the potential user with an overview of the capabilities and structural design of the GTS system. Successive volumes provide a more detailed description of the various components of the GTS system and a description of the input data required to execute the software.

The remaining volumes of the documentation series are:

- Volume II. GTS Usage Guide
- Volume III. GTS Flight Dynamics Models
- Volume IV. GTS Numerical Operators
- Volume V. GTS Weight Estimation Models for Sizing Applications

The Generalized Trajectory Simulation (GTS) system comprises a linked set of digital computer programs used for many Aerospace Corporation boost, satellite, and reentry vehicle/mission simulation studies. GTS satisfies the requirements for trajectory simulations, vehicle sizing, and optimization capabilities beyond those available in other computational tools. Architecturally, the software system exploits the logical concept of complete separation of mathematical operations, physical modeling, and software control functions. The system software is completely programmed, with some minor exceptions, in a FORTRAN language compatible with CDC 6000/7000 series computers. Applications may be submitted to the system via the normal card reader input queue or, alternately, from remote terminals.

The primary design consideration for the GTS system has been the provision of an easy-to-use software tool that permits a variety of users, with diverse applications, to individually balance their requirements for completeness and accuracy with cost and time constraints. Consequently, user-oriented input data specification, computational efficiency, wide applicability, and straightforward software development and modification features have been primary objectives in the design of the program. These objectives have been accomplished, in part, by carefully separating

the mathematical operators, the engineering models, and the executive program control functions. The logic governing the interconnection between program components is under input processing control. As a result of this design, the user's input data is processed and related to a directory of GTS subroutines so that the minimal software configuration required for the particular application is automatically loaded and executed.

The modular design of GTS provides an exceptionally convenient means by which modifications or extensions of the existing capabilities can be implemented, validation and verification of new models accomplished, and overall operating efficiency maintained at a high level. These features are of particular value in those applications which are exploratory in nature. The major components of the GTS system are given in Table 1.

1.1 GTS Operational Capabilities

The major functions and capabilities of the GTS system are summarized in this section to provide a convenient introductory reference. Each of the significant capabilities is briefly described in subsequent paragraphs of this section.

1.1.1 Trajectory Simulation

The GTS trajectory simulation capability provides a basic framework to define and control a diversity of vehicle and mission simulations. The user may define new physical models or select models from an existing large library of flight dynamic models. A complete description of the existing models is provided in Volume III. The GTS trajectory control feature allows a wide degree of flexibility for the specification of required flight profiles and mission objectives. General areas of application of the trajectory simulation capability are indicated in Table 2.

Table 1. Principal Components of the GTS System

- A SIMULATION LIBRARY OFFERING A LARGE SELECTION OF MATHEMATICAL OPERATORS AND PHYSICAL MODELS REQUIRED FOR THE ANALYSIS OF VEHICLE MISSION SYSTEMS
- THE GENERALIZED TRAJECTORY LANGUAGE PREPROCESSOR (GTL), WHICH SCANS AND INTERPRETS THE USER-ORIENTED INPUT DATA, EXTRACTS AND REFORMULATES MANY IMPLIED FUNCTIONS, AND GENERATES THE GTS MODEL DATA REQUIREMENTS FOR THE PARTICULAR PROBLEM
- A PREPROCESSOR PROGRAM WHICH CORRELATES THE MODEL-DATA REQUIREMENTS PRODUCED BY GTL WITH A DETAILED DIRECTORY OF THE PROGRAM'S SUBROUTINE INTERCONNECTIONS AND FROM THIS GENERATES THE REQUIRED PROGRAM CONFIGURATION FOR THE PARTICULAR PROBLEM
- GTS MASTER EXECUTIVE WHICH EXECUTES THE REQUIRED PROGRAM CONFIGURATION, USING THE APPROPRIATE PROCESSING CAPABILITIES TO SOLVE THE PARTICULAR PROBLEM

Table 2. Representative Trajectory Simulation Applications

- | | |
|--|--|
| • VEHICLE CAPABILITY ANALYSIS | • TUMBLING TURN ANALYSIS |
| • BOOSTER SELECTION | • LOW THRUST ORBITAL MANEUVERS |
| • MISSION REFERENCE TRAJECTORIES | • ABORT PROPULSION REQUIREMENTS |
| • LAUNCH WINDOW STUDIES | • DE-ORBIT PROPULSION REQUIREMENTS |
| • VEHICLE STABILITIES STUDIES | • RENDEZVOUS MISSION STUDIES |
| • LAUNCH SUPPORT | • FLIGHT SAFETY SYSTEMS |
| • GUIDANCE EQUATION DEVELOPMENT | • TERMINAL MANEUVER ANALYSIS |
| • LIFTING BOOST VEHICLE ANALYSIS | • ASCENT AND ORBITAL ABORT STUDIES |
| • RECOVERABLE BOOSTER STAGING | • FILTER RADAR TRACKING DATA |
| • BALLISTIC AND MANEUVERING REENTRY VEHICLES | • SATELLITE SYSTEM ANALYSIS |
| • AIRCRAFT TRAJECTORY SIMULATION | • CONTROL SYSTEM STUDIES |
| • ERROR ANALYSIS | • ORBITAL TRANSFER OPTIMIZATION |
| • POSTFLIGHT RECONSTRUCTION | • ORBIT LIFETIME STUDIES |
| • SOFTWARE VALIDATION | • DEBRIS DISPERSION STUDIES |
| • SOFTWARE VERIFICATION | • INTERPRETIVE COMPUTER SIMULATION (ICS) |

1.1.2 Optimization

The GTS system includes a general purpose nonlinear constrained optimization capability that can be utilized independently or in conjunction with the trajectory simulation and vehicle sizing capabilities. The optimization capability is designed to optimize (i. e. , maximize or minimize) an objective function, $f(x)$, where x is a vector, subject to a set of nonlinear equality and inequality constraints. The operator is mechanized to utilize analytic partial derivatives, if available; otherwise numerical partial derivatives are computed. The optimization features of the GTS system are described, in detail, in Volume IV.

1.1.3 Vehicle Sizing Analysis

The GTS vehicle sizing models provide a vehicle sizing capability for space and missile vehicles that utilize solid propellant rocket motors. Liquid propelled vehicles can also be accommodated within the sizing procedure. They, however, require the development of specialized weight estimation models to represent particular characteristics of the desired liquid propellant system.

The sizing models estimate performance sensitive component weights to determine a vehicle configuration consistent with realistic geometry and mission constraints.

Intended applications include preliminary design studies, booster subsystem trade-off studies, and growth studies of existing systems including the analysis of advanced propellant technology or new launching concepts. The existing library of weight estimation models is completely documented, including information on their use and interface with other capabilities of the GTS system, in Volume V.

1.1.4 Generalized Trajectory Language (GTL)

The input to all of the operational capabilities of the GTS system is via a higher level symbolic input language especially designed to accom-

modate trajectory simulations, vehicle sizing, and optimization applications. GTL not only controls the input of data to the program, it also is used to select and build the actual software configuration that is loaded into the computer and executed.

GTL provides a free form, user-oriented input method that is straightforward to learn and use. This approach encourages concentration on the input requirements of the engineering application with little, or no, concern for the intricate details of the software. The input language also allows a user to include FORTRAN subroutines as a portion of the input data deck. These routines are processed at program execution time and utilized in a fashion similar to any other model within the GTS system. The GTL language is described in Volume II.

SECTION 2

GTS SYSTEM CHARACTERISTICS

A description of the characteristics of the major GTS system capabilities is presented in this section. A detailed description of these capabilities, including a discussion of the input data required to apply these capabilities to specific problems, is presented in the remaining volumes of the GTS documentation series.

2.1 Trajectory Applications

The trajectory simulation capability of the GTS system is applicable to a variety of vehicle configurations, mission trajectory profiles, and mission objectives, as indicated by Table 2. Moreover, the modular structure of the GTS system enables the user to satisfy his particular requirements for efficiency and completeness. The basis of the trajectory simulation capability is a set of flight dynamic models. The organization of these models typifies the modular design of GTS. This organization is discussed in the next section. An overview of the manner by which these individual models are incorporated into a trajectory simulation is then discussed.

2.1.1 Model Types and Models

The flight dynamics models are organized into functional units referred to as model types. Associated with each model type is a set of models. Each of the models associated with a specific model type simulates a particular function or element of the flight dynamics. For example, an atmosphere model simulates the atmosphere, or a weight model computes the vehicle weight. A complete list of the trajectory simulation model types and their corresponding simulation function is presented in Table 3. Clearly, there is no unique way to simulate these functions. Moreover, different applications require varying levels of complexity of the functions being simulated. Consequently, a variety of models are available within each model type. For example, there are currently 7 different atmospheric models available. That is, each model within a model type simulates the same engineering function but in a different or possibly more sophisticated

Table 3. Trajectory Simulation Model Types in GTS

MODEL TYPE MNEMONIC	FUNCTION
INIT	INITIALIZE VEHICLE POSITION AND VELOCITY
GRAV	SIMULATE ACCELERATION DUE TO THE ATTRACTION OF THE CENTRAL BODY
ATM	SIMULATE ATMOSPHERE
WIND	SIMULATE WINDS
WEIGHT	COMPUTE VEHICLE WEIGHTS
GUID	COMPUTE GUIDANCE COMMANDS, OR SIMULATE AIRBORNE COMPUTER
CONTROL	SIMULATE AUTOPILOT OR OPEN LOOP STEERING
ATTITUDE	COMPUTE BODY ATTITUDE
AEROF	COMPUTE AERODYNAMIC FORCES
AEROM	COMPUTE AERODYNAMIC MOMENTS
AERO	COMPUTE AERODYNAMIC FORCES AND MOMENTS
ACTUATOR	SIMULATE ACTUATORS
ENGINE	SIMULATE ENGINE
PROPF	COMPUTE PROPULSIVE FORCE VECTOR
PROPM	COMPUTE PROPULSIVE MOMENTS
TEOM	INTRODUCE THE TRANSLATIONAL EQUATIONS OF MOTION
REQM	INTRODUCE THE ROTATIONAL EQUATIONS OF MOTION
SENSOR	SIMULATE SENSORS: e.g., ACCELEROMETERS, GYROS, PLATFORMS, IMUs
DER	INTEGRATE AUXILIARY EQUATIONS OF INTEREST (e.g., velocity losses)
NONDER	COMPUTE AUXILIARY QUANTITIES OF INTEREST

manner. Thus, the user is able to select the model that realistically and efficiently represents his problem.

For a particular trajectory simulation, the software configuration that is actually executed includes only those model types and associated models which the user specified for that simulation. At user-specified events new data may be introduced into the simulation by changing the data associated with a given model, by changing the model associated with a given model type, or by introducing new model types and associated models.

2.1.2 Trajectory Profile Specification

Within the GTS system, trajectory simulations are performed by describing the desired trajectory profile as a series of phases separated by events. A trajectory is defined to be the entire vehicle state-time history of interest. A discrete time point along a trajectory is referred to as an event. The user may specify conditions which define events. These conditions usually refer to an instantaneous state of the vehicle as computed by the program. Examples of event criteria are:

- (i) flight path angle equal to -15 degrees
- (ii) vehicle altitude increasing and equal to 300000 feet
- (iii) maximum value of dynamic pressure.

Several actions may result from the detection of an event. For example, new data may be introduced to the simulation which will alter the course of the trajectory, or information may be output which will leave the course of the trajectory unaltered.

A phase is defined to be a continuous portion of a trajectory between two successive event occurrences. Thus, a phase is always initiated and terminated by an event. Events are of particular interest to the user, since it is by the response to user-specified events that new data is introduced to the simulation or that the trajectory is terminated. Precise information concerning the types of events and the specification of event criteria is presented in Volume II.

2.2 Sizing Applications

The GTS system includes a set of weight estimation models which, combined with the GTS optimization and trajectory simulation features, form a generalized sizing capability for vehicles utilizing solid propellant rocket motors. Although ultimate component design is not precluded, the sizing capability is normally for preliminary design studies to estimate component weights which will determine a propulsion system configuration consistent with realistic vehicle geometry, performance, and mission constraints. Parametric weight scaling equations are utilized for sizing each major rocket component. The sizing capability does not require the specification of reference designs prior to generating results and provides for the inclusion of technology changes. Results are valid for propulsion system weights between 3,000 and 2,000,000 pounds.

The sizing capability utilizes a model type-model structure similar in nature to the trajectory simulation capability. It may be executed as a stand-alone application or in conjunction with the trajectory simulation. For example, the sizing capability may be executed to evaluate a vehicle geometry, weight, and propulsion quantities. Subsequently, at the user's option, vehicle parameters computed by the sizing application are available as input data for a trajectory simulation. The optimization process is invoked to satisfy the required vehicle geometry constraint.

Note that the aforementioned application of the optimization capability does not require any trajectory simulations. After the optimized vehicle configuration has been determined, a trajectory simulation utilizing the optimized vehicle may be requested.

A variation of the procedure just described is the application of the optimization operator to the satisfaction of coupled vehicle/trajectory constraints. In this instance, the objective might be the determination of a set of vehicle parameters that satisfy a set of vehicle geometry constraints and maximizes the range of the vehicle. For such an

application, the sizing capability, the trajectory simulation capability and the optimization capability are all executed as a part of the same application. Additional information on the individual sizing models and examples of the use of the sizing capability are presented in Volume V.

2.3 Numerical Operators

The extensive GTS library of physical and engineering models is supplemented with a compatible set of numerical operators. These operators are distinguished from the physical and engineering models in that they provide methods for the solution of mathematical problems associated with scientific and engineering simulations. The three major categories of numerical operators available within the GTS system are; (1) integration operators, (2) optimization and search operators, and (3) interpolation operators. Analogous to the engineering operators, GTS provides a choice of operators in each category. Furthermore, the numerical operators are independent of the engineering models in the sense that any of the numerical operators may be specified in conjunction with any particular model configuration the user has selected. For example, any one of the available integration operators may be selected for a particular trajectory simulation. Similarly, any one of the available optimization algorithms may be executed as a part of the trajectory optimization capability or vehicle sizing capability. Consequently, the engineering analyst may choose the numerical algorithms which provide the desired balance between accuracy and efficiency for his particular problem. These numerical algorithms, along with the associated input data, are described in Volume IV.

2.3.1 Integration Techniques

While the integration techniques in GTS are general purpose, these techniques have been chosen with particular attention to their viability in

solving sets of differential equations associated with trajectory simulations. Variations of mission profiles, accuracy requirements, and computational efficiency dictate that several techniques be available. These techniques are Runge-Kutta fixed step (fourth order), Adams-Moulton fixed step (m-th order, $1 \leq m \leq 8$), and Adams-Moulton variable step, (m-th order, $1 \leq m \leq 8$). The integration operators are not restricted to a pre-set number of differential equations. The number of differential equations is dependent upon the particular simulation, and typically varies from as few as six to over one hundred equations.

2.3.2 Optimization Capability

The GTS system contains two state-of-the-art nonlinear constrained parameter optimization operators. They are designed to solve the following problem. Determine the values of the n parameters, or independent variables, x_1, x_2, \dots, x_n , such that the scalar function $f(x) = f(x_1, x_2, \dots, x_n)$ is optimized (maximized or minimized), subject to equality constraints of the form

$$c_i(x) = 0 \quad , \quad i = 1, 2, \dots, k$$

and inequality constraints of the form

$$c_i(x) \geq 0 \quad , \quad i = k+1, \dots, m$$

Special cases of the above problem include constraint solving problems in which there is no objective function and the number of variables is greater than or equal to the number of constraints, and nonlinear least squares problems in which there is no objective function and the number of constraints is greater than the number of variables. Also, optimal control problems may be solved using the optimization techniques available in GTS.

These optimization and search techniques are fully incorporated into the GTS system. The complete library of physical models is available as part of the definition of an optimization problem. The input required to define optimization problems is compatible with the input required to define

trajectory simulation and vehicle sizing problems. Thus, the definition of the optimization problem does not require a separate specification of the associated trajectory simulation or weight estimation problem. To illustrate, assume that a user has defined a trajectory simulation, and as a part of further analysis, he wishes to define an associated trajectory optimization problem. The only additional input required is a specification of which trajectory input quantities are to be the independent variables and which computed quantities are to be the objective function and/or constraints. In this regard, any input quantity may be an optimization variable. Thus, for example, an independent variable may be a tabular value, a model input value, or a quantity which is part of an event specification. Likewise, the user has complete freedom in choosing which quantities form the objective function and constraints.

Furthermore, the independence of the physical models and optimization provides the user the capability to formulate many diverse types of problems. As indicated, a variety of trajectory optimization problems may be formulated, such as boost trajectory problems, reentry problems, or orbital transfer problems. Likewise, vehicle sizing applications may be defined which do not require trajectory simulations, or vehicle sizing applications may be specified which require both the vehicle sizing and trajectory simulation capabilities. Finally, the operators may also be used to obtain the optimal solution to a large class of problems having similar mathematical characteristics to trajectory and vehicle sizing problems.

Some of the optimization algorithms utilize partial derivative information. If the information can be computed analytically, the optimization operator will effectively use this information. Otherwise, a scheme for obtaining numerical partials is implemented.

2.4 General Applicability of the GTS System

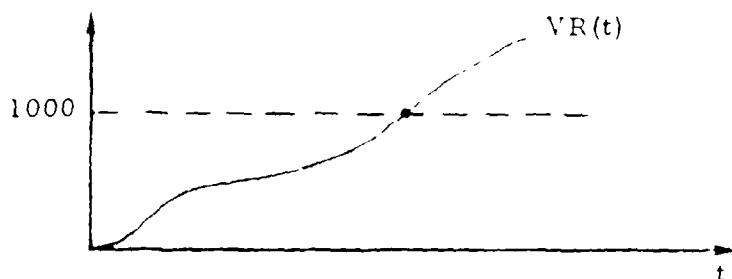
Many of the capabilities of the GTS system have applicability to areas other than trajectory simulation or vehicle design. The numerical operators, for instance, provide numerical methods for solving mathematical problems

which may arise independent of any vehicle design or trajectory simulation considerations. For example, the integration operators provide a capability for the solution of any set of first order, initial value differential equations. Likewise, the optimization operators are designed to solve a general non-linear parameter optimization problem. The flexibility of the GTS input characteristics permits these operators to be properly invoked to solve the particular problem of interest. This capability is illustrated in Volume II.

2.5 Generalized Trajectory Language (GTL)

GTL is a higher level input language included as an integral part of the GTS system. GTL interprets card images and formulates all the input into a form acceptable to the internal GTS input processor. GTL provides a user-oriented format that permits a user to define the application (trajectory, vehicle sizing, optimization, etc.), the individual models required, the associated numerical operators, and all of the required input data values in a free form mnemonic oriented fashion. A complete description of GTL is part of Volume II.

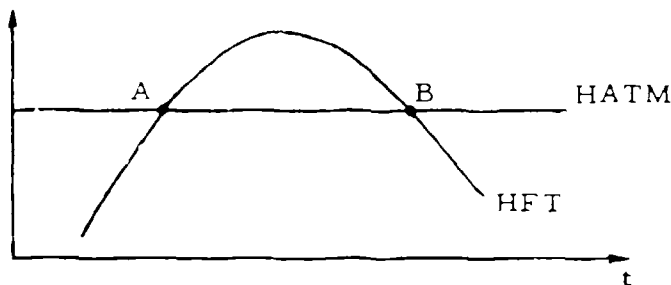
To illustrate GTL, consider the following examples of GTL data specification. Relative to the trajectory simulation capability, the following examples illustrate the specifications of events in GTL. Assume an event occurs when the criterion that the variable VR is equal to 1000 is satisfied.



The GTL specification, that is the actual input that users must supply, for this event is

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	

Next, assume an event occurs when the criterion that the variable HFT is equal to the variable HATM, and HFT is increasing (point A on the graph).



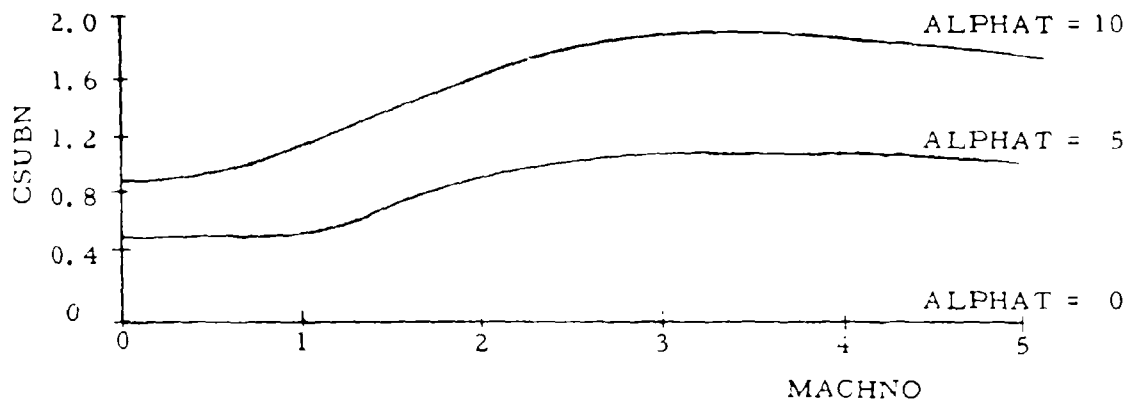
The GTL specification for this event is

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37		
						HFT[+] = HATM																																

Alternatively, if an event is to occur when HFT is equal to HATM and HFT is decreasing (point B on the graph), then the GTL specification of this event is

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	

A common method for specifying functional relationships in GTS is via tabular data. GTL provides a convenient format for tabular input. As an illustration, assume that the quantity CSUBN is a function of two independent variables, ALPHAT and MACHNO, as represented in the following graph.



The corresponding GTL tabular input, assuming quadratic interpolation, is the following.

NORMAL FORCE COEFFICIENT TABLE			
ORDER = 2			
ALPHAT = 0			
MACHNO	CSUBN	MACHNO	CSUBN
0	0	1.25	0
.50	0	1.75	0
.75	0	2.75	0
1.00	0	5	0
ALPHAT = 5			
MACHNO	CSUBN	MACHNO	CSUBN
0	.44	1.25	.64
.50	.49	1.75	.76
.75	.52	2.75	.86
1.00	.56	5	.84
ALPHAT = 10			
MACHNO	CSUBN	MACHNO	CSUBN
0	.96	1.25	1.26
.50	.98	1.75	1.52
.75	1.00	2.75	1.76
1.00	1.10	5	1.74

In order to provide a convenient method for executing the optimization capability, GTL includes an optimization input format. The format provides the user with an easy-to-use mechanism for specifying optimization input data. For example, the choice of the optimization operator is specified by a statement of the form

20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60																			
																																								THE OPTIMIZATION OPERATOR IS UOPTIM																			

A constraint that the variable VR, evaluated at event PO50, be greater than or equal to 1000 is specified by

41	5	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53
										(VR.AT.PO50 ≥ 1000.)																													

A constraint that the variable GDLV be equal to GDLPD plus 30, is specified by

41	5	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53
										(GDLV = GDLPD + 30.)																													

An additional GTL capability allows user coded FORTRAN routines to be included in the GTL input data. These routines are automatically inserted into the GTS simulation. Thus, the user can request that unique auxiliary computations be made or that minor modifications to existing GTS models be made without modifying the existing GTS models.

4.6 Input/Output Capabilities

The GTS system allows the user to specify his input data and receive his output data in various forms and formats. The primary means of specifying input data is by means of punched cards. However, the user may

access input data which resides on a disk file or magnetic tape. Special purpose input data, such as radar tracking data, may be input to a problem. The system requires the user to describe the format of the input data. That is, the number of words per data frame or the relative locations of the particular data items of interest are specified to the system via GTL input.

GTS has the capability of producing several types of output all under the control of the user. Included are printed output in a standard format, printed output and/or case summary print which is formatted by the user, binary data files for subsequent use in other software tools, and graphical output processors.

The standard output print (block print) follows the model types of GTL input data. That is, the printed data is formed into distinct blocks where each block is associated with a particular model type (e.g., AEROF, CONTROL). A data item may be a floating point number, an integer number, a logical value, or an alpha-numeric character string. The user may request that blocks be edited to reduce the output or that data items which are not included in a standard print block be printed in a supplementary data block. A block print key is printed at the beginning of each case.

User-defined output is handled by a special output operator. Any GTS variable or user-defined variable name is an acceptable output quantity. Multiple executions of the output operator are permissible, and each call may reference a different format statement and list of variables. In addition, if logical tests are required to determine what to print or when to print, a subroutine may be written and input with the GTL input data to perform these functions.

Plots are obtained by writing the output data to be plotted on files using the special output model. The plotting capabilities are executed as post-processors and are accessed by means of GTL input. Also available is a world map plotting capability. The user may also request that any or all input data tables (interpolation or step tables) be plotted.

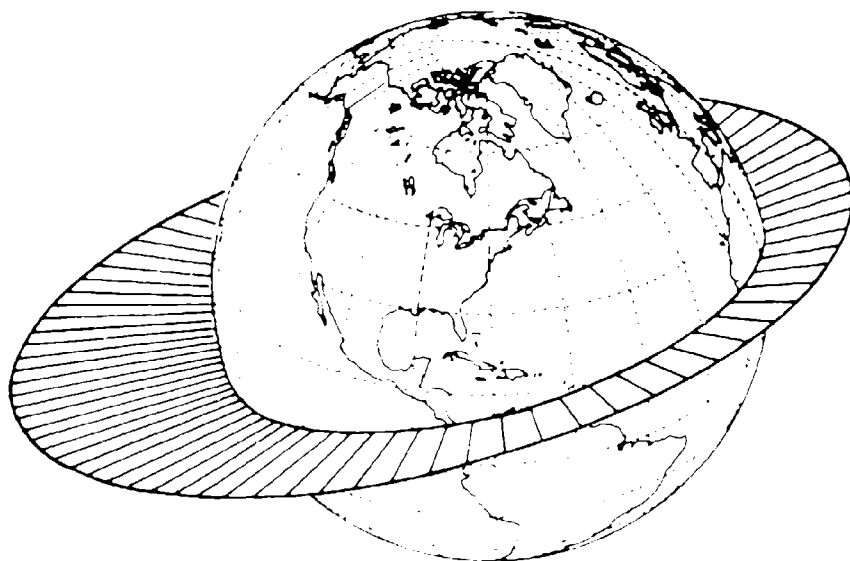
2.7 Trajectory Simulation Example

As an illustration of the use of the GTS system, consider the simulation of the three degree-of-freedom trajectory for a satellite in orbit. The data specified to define the orbit and establish the initial position of the vehicle are:

apogee altitude	=	3000 nautical miles
perigee altitude	=	1000 nautical miles
inclination of orbit	=	20°
argument of perigee	=	85°
true anomaly	=	260°
satellite longitude	=	-119°

A spherical non-rotating earth model is used in this example. The integration method is 8th order, variable step, Adams-Moulton. The initial integration interval is 8 seconds. Trajectory data is to be printed at apogee and perigee (i. e., whenever the inertial flight path angle is zero).

The initial configuration of the orbit is shown in Figure 1. The data as it appears on a GTS coding form is shown in Figure 2. A computer generated listing of the CTL input and trajectory output is also included. Additional examples which illustrate the capabilities of the GTS system are detailed in Volume II.



ν - TRUE ANOMALY
 U_P - ARGUMENT OF PERIGEE

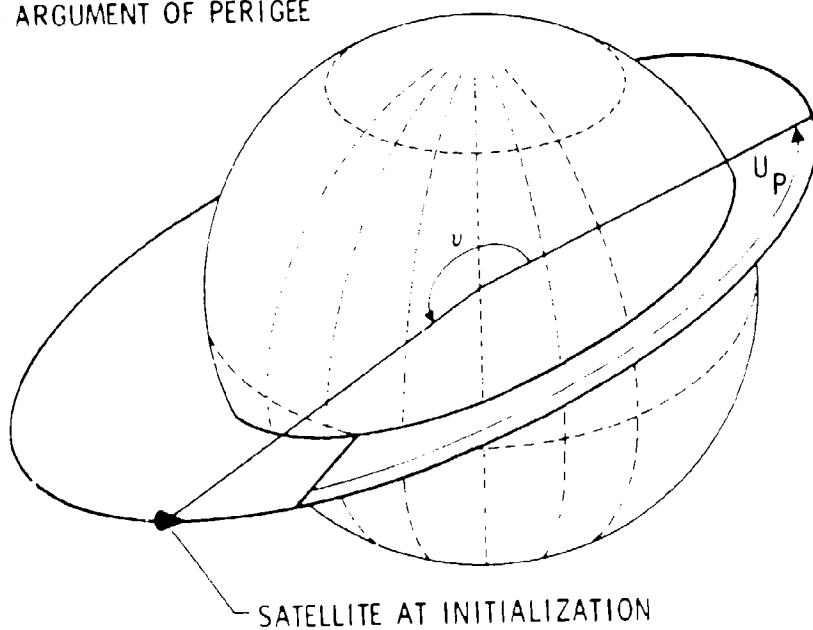


Fig. 1. Orbit Configuration

GTS INPUT FORM

THE AEROSPACE CORPORATION

PROGRAMMER	KEYPUNCHED	VERIFIED	DATE	PAGE	OF
1000 ORBIT ECCENTRIC ORBIT					
1001 BEGIN EVENT SEQUENCE					
1002 P010					
1003 PH10 INITIALIZE					
1004 INIT					
1005 VEHICLE ON ORBIT					
1006 INITM30					
1007 HAR					
1008 = 3000					
1009 MPD					
1010 = 1000					
1011 INCLIN					
1012 = 20					
1013 UP					
1014 = 85					
1015 TANMIN					
1016 = 260					
1017 LTP01					
1018 = -119					
1019 GRAY					
1020 GRVMI					
1021 CACON					
1022 = 2					
1023 ROTATE					
1024 = 0					
1025 TEGMI					
1026 INTGRM1					
1027 METHOD					
1028 = 3AM					
1029 VARIABLE STEP:					
1030 ORDER					
1031 = 8					
1032 DTIN					
1033 = 8					
1034 PUT					
1035 PUTPUT					
1036 REZO GAMMA1					
1037 = 0					
1038 P0100					
1039 TIME = 12000					
1040 STOP/					
1041 END EVENT SEQUENCE					
1042 END OF CASE					

Fig. 2. GTL Input Data

ENVIRONMENTAL DATA
 CENTRAL BODY SPHERICAL
 NON-ROTATING
 EQUATORIAL RADIUS
 .20902900E+08
 POLAR RADIUS
 .20302900E+08
 ATMOSPHERE NO ATMOSPHERE
 ROTATION RATE
 0.
 GRAV. CONSTANT
 .14076519E+17
 PRESSURE AT SEA LEVEL
 0.

BLOCK PRINT KEY

TIME	01NOH	VI	WFT	M	GAMMAI	GCLV	RV	AZI	1	GOLV	LV	OW	A7R
XECI	DOXECI	GEI(1)	YECI	DOYECI	GEI(2)	DOXECI	DOYECI	GEI(3)	4	DOXECI	DOYECI	DOXECI	DOYECI
RANGRLS	RANGRLS	RANGRLS	RANGRLS	RANGRLS	RANGRLS	RANGRLS	RANGRLS	RANGRLS	7	ILV	CRNGRF	CRNGRF	CRNGRF
ORBIT TYPE	INCNODE	INCCLV	ECC	TRANOMV	WAPOG	WAPOG	WAPOG	WAPOG	11	RANGIIP	CRNGRF	CRNGRF	CRNGRF
HA	HP	HP	HP	HP	HP	HP	HP	HP	30	KEAPE	OLMODE	SWAORR	SWAORR
									31	PERIOD	LNODE	MUNODE	MUNODE
									32	TNODE	LVA	RA	RA
									33	GOLVA	LVA	QP	QP
									34	GOLVA	LVA		
									35	TPERG	MUPERG		

The preceding BLOCK PRINT KEY displays print key lines which contain mnemonic program symbols. The print key lines are identified by line numbers. These line numbers are used to relate the lines of printed output in the BLOCK PRINT with the print key lines contained within the BLOCK PRINT KEY.

Brief definitions of the output quantities are presented below. The output quantities are identified by their mnemonic program symbol, and are ordered by line number.

LINE 1

TIME	Current time within the dynamic system
HFT	Altitude of the vehicle (feet)
GCLV	Geocentric latitude of the vehicle
GDLV	Geodetic latitude of the vehicle
LV	Longitude of the vehicle
W	Total vehicle weight

LINE 2

DTNOM	Nominal integration step size
H	Altitude of the vehicle (nautical miles)
RV	Radial distance from center of earth to vehicle
ETATOT	Total load factor

DW	Total weight flow-rate from vehicle

LINE 3

VI	Magnitude of the inertial velocity vector
GAMMAI	Inertial flight path angle
AZI	Inertial azimuth
VR	Magnitude of the earth relative velocity vector
GAMMAR	Earth relative flight path angle
AZR	Earth relative azimuth

LINE 4

XECI	The ECI X coordinate of vehicle position
YECI	The ECI Y coordinate of vehicle position
ZECI	The ECI Z coordinate of vehicle position
DXECI	Time derivative of XECI
DYECI	Time derivative of YECI
DZECI	Time derivative of ZECI

LINE 5

DDXECI	Time derivative of DXECI
DDYECI	Time derivative of DYECI
DDZECI	Time derivative of DZECI
SDDXECI	The ECI X component of sensed acceleration
SDDYECI	The ECI Y component of sensed acceleration
SDDZECI	The ECI Z component of sensed acceleration

LINE 7

GECI(1)	The ECI X component of gravitational acceleration
GECI(2)	The ECI Y component of gravitational acceleration
GECI(3)	The ECI Z component of gravitational acceleration
ILV	Inertial longitude of the vehicle

LINE 11

RANGRLS	Relative range from launch site to current position.
RANGRIP	Relative range from initial position to current position
RANGILS	Inertial range from launch site to current position
RANGIIP	Inertial range from initial position to current position

LINE 30

ORBIT	
TYPE	Type of orbit
SOVRCEA	Ratio of semilatus-rectum to equatorial radius
KE/PE	Ratio of kinetic energy to potential energy

LINE 31

INCNODE	Orbit inclination at next ascending node
ECC	Orbit eccentricity
DOMEGAP	Inertial motion of apsides
PERIOD	Orbit period
DLNODE	Inertial motion of ascending node
SMAORB	Orbit semimajor axis

LINE 32

INCLV	Local orbit inclination
TRANOMV	True anomaly
OMEGAV	Argument of latitude of current position
TNODE	Time of next ascending node
LNODE	Relative longitude of next ascending node
MUNODE	Inertial range angle from present position to next ascending node

LINE 33

HA	Altitude of next apogee
VAPOG	Inertial velocity at next apogee
OMEGAA	Argument of latitude of next apogee
GDLVA	Geodetic latitude at next apogee
LVA	Relative longitude at next apogee
RA	Radial distance from the center of the earth to the next apogee

LINE 34

HP	Altitude at next perigee
VPERG	Inertial velocity at next perigee
OMEGAP	Argument of latitude of next perigee
GDLVP	Geodetic latitude at next perigee
LVP	Relative longitude at next perigee
RP	Radial distance from the center of the earth to the next perigee

LINE 35

TAPOG	Time of next apogee
MUAPOG	Inertial range angle from present position to next apogee
TPERG	Time of next perigee
MUPERG	Inertial range angle from present position to next perigee

1941

BEGIN NEW PHASE -- INITIALIZE VEHICLE ON ORBIT

ORBIT TYPE	TIME	0.000000	---	---	---	---	---
0	12080402.9	-5.07056582	1	-5.07054582	-119.000000	0	0
0.00000000	1989.449525	3299.1022.9	3	26676.0280	-10.5916336	70.6399448	0
20016.0200	-10.5915336	70.6299448	4	18315.3498	-6506.44030	7050.69694	0
-1591711.1	-20741567.7	-2920433.91	5	-119.000000	0	0	0
6.245339504	1.1.2669909	1.14484021	11	244266599E-10	0	0	0
6.245339584	1.1.2669909	7132.23375	30	200965529	0	330551331.0	0
7132.52375	2.244266599E-10	1.52793315	31	160.740910	-104.067277	15.0000000	0
ELIPIIC	1.52793315	0	32	444.9613399	-159.013219	19131246.5	0
20.00000000	1.52793315	0	33	395.000000	-20.211863006	26979015.5	0
300.00000000	2.9713.02286	395.000000	34	444.9613399	-104.067277	15.0000000	0
300.00000000	2.9713.02286	395.000000	35	395.000000	-20.211863006	26979015.5	0
1000.000000	7237.000000	395.000000	36	444.9613399	-104.067277	15.0000000	0

EVENT	EVENT	EVENT	EVENT
1. The first event is the start of the race.	2. The second event is the first lap.	3. The third event is the second lap.	4. The fourth event is the third lap.
5. The fifth event is the fourth lap.	6. The sixth event is the fifth lap.	7. The seventh event is the sixth lap.	8. The eighth event is the seventh lap.
9. The ninth event is the eighth lap.	10. The tenth event is the ninth lap.	11. The eleventh event is the tenth lap.	12. The twelfth event is the eleventh lap.
13. The thirteenth event is the twelfth lap.	14. The fourteenth event is the thirteenth lap.	15. The fifteenth event is the fourteenth lap.	16. The sixteenth event is the fifteenth lap.
17. The seventeenth event is the sixteenth lap.	18. The eighteenth event is the seventeenth lap.	19. The nineteenth event is the eighteenth lap.	20. The twentieth event is the nineteenth lap.
21. The twenty-first event is the twentieth lap.	22. The twenty-second event is the twenty-first lap.	23. The twenty-third event is the twenty-second lap.	24. The twenty-fourth event is the twenty-third lap.
25. The twenty-fifth event is the twenty-fourth lap.	26. The twenty-sixth event is the twenty-fifth lap.	27. The twenty-seventh event is the twenty-sixth lap.	28. The twenty-eighth event is the twenty-seventh lap.
29. The twenty-ninth event is the twenty-eighth lap.	30. The thirtieth event is the twenty-ninth lap.	31. The thirty-first event is the thirtieth lap.	32. The thirty-second event is the thirty-first lap.
33. The thirty-third event is the thirty-second lap.	34. The thirty-fourth event is the thirty-third lap.	35. The thirty-fifth event is the thirty-fourth lap.	36. The thirty-sixth event is the thirty-fifth lap.
37. The thirty-seventh event is the thirty-sixth lap.	38. The thirty-eighth event is the thirty-seventh lap.	39. The thirty-ninth event is the thirty-eighth lap.	40. The fortieth event is the thirty-ninth lap.
41. The forty-first event is the fortieth lap.	42. The forty-second event is the forty-first lap.	43. The forty-third event is the forty-second lap.	44. The forty-fourth event is the forty-third lap.
45. The forty-fifth event is the forty-fourth lap.	46. The forty-sixth event is the forty-fifth lap.	47. The forty-seventh event is the forty-sixth lap.	48. The forty-eighth event is the forty-seventh lap.
49. The forty-ninth event is the forty-eighth lap.	50. The fiftieth event is the forty-ninth lap.	51. The fifty-first event is the fiftieth lap.	52. The fifty-second event is the fifty-first lap.
53. The fifty-third event is the fifty-second lap.	54. The fifty-fourth event is the fifty-third lap.	55. The fifty-fifth event is the fifty-fourth lap.	56. The fifty-sixth event is the fifty-fifth lap.
57. The fifty-seventh event is the fifty-sixth lap.	58. The fifty-eighth event is the fifty-seventh lap.	59. The fifty-ninth event is the fifty-eighth lap.	60. The sixtieth event is the fifty-ninth lap.
61. The sixty-first event is the sixtieth lap.	62. The sixty-second event is the sixty-first lap.	63. The sixty-third event is the sixty-second lap.	64. The sixty-fourth event is the sixty-third lap.
65. The sixty-fifth event is the sixty-fourth lap.	66. The sixty-sixth event is the sixty-fifth lap.	67. The sixty-seventh event is the sixty-sixth lap.	68. The sixty-eighth event is the sixty-seventh lap.
69. The sixty-ninth event is the sixty-eighth lap.	70. The seventieth event is the sixty-ninth lap.	71. The seventy-first event is the seventieth lap.	72. The seventy-second event is the seventy-first lap.
73. The seventy-third event is the seventy-second lap.	74. The seventy-fourth event is the seventy-third lap.	75. The seventy-fifth event is the seventy-fourth lap.	76. The seventy-sixth event is the seventy-fifth lap.
77. The seventy-seventh event is the seventy-sixth lap.	78. The seventy-eighth event is the seventy-seventh lap.	79. The seventy-ninth event is the seventy-eighth lap.	80. The eightieth event is the seventy-ninth lap.
81. The eighty-first event is the eightieth lap.	82. The eighty-second event is the eighty-first lap.	83. The eighty-third event is the eighty-second lap.	84. The eighty-fourth event is the eighty-third lap.
85. The eighty-fifth event is the eighty-fourth lap.	86. The eighty-sixth event is the eighty-fifth lap.	87. The eighty-seventh event is the eighty-sixth lap.	88. The eighty-eighth event is the eighty-seventh lap.
89. The eighty-ninth event is the eighty-eighth lap.	90. The ninetieth event is the eighty-ninth lap.	91. The ninety-first event is the ninetieth lap.	92. The ninety-second event is the ninety-first lap.
93. The ninety-third event is the ninety-second lap.	94. The ninety-fourth event is the ninety-third lap.	95. The ninety-fifth event is the ninety-fourth lap.	96. The ninety-sixth event is the ninety-fifth lap.
97. The ninety-seventh event is the ninety-sixth lap.	98. The ninety-eighth event is the ninety-seventh lap.	99. The ninety-ninth event is the ninety-eighth lap.	100. The hundredth event is the ninety-ninth lap.

```

      SSC = RE2D      ) TYPE = 0
      THIS EVENT OCCURRED BY SATISFACTION OF THE CONDITION -- GAMMAI
      = -.354419266643E-07

```

[illegible]

EVENT	EVENT	EVENT	EVENT	EVENT	EVENT
ESC = REZ (REZ) TYPE = -0 THIS EVENT OCCURRED BY SATISFACTION OF THE CONDITION -- GAMPAI = .175110667205E-06					
TIME = 7237.0243697					
2317-02437	10227459-0	-19-9206500	1	-19-9206500	0
256-000000	2009-91970	39130759-0	2	39130759-0	0
17134-9958	-175110667E-06	91-0170971	3	17134-9958	91-0170971
-345220433	12695391-5	-13132520-3	4	-5736-27033	-510-023633
8-11200338	-2-9825616	3-11323947	5	159-013137	0
9116-09010	4000-02720	9116-09010	11	4000-02720	0
ORBIT TYPE	ELLIPTIC	1-52792526	30	400094166	33054005-4
20-000000	1383811668	1-52792526	31	1677330641	950000533
20-000000	179-999999	264-999547	32	104693000	19130759-0
2999-91970	17134-9958	264-999547	33	-10-9206500	26979011-8
999-999995	24852-0521	444-999547	34	10-9206500	
	7237-02440	.777586894E-06	35	12269-1956	
TIME = 12000.0000000					
12000-0000	6204201-20	10-9405017	1	10-9405017	0
256-000000	1021-00321	27107101-3	2	0	0
24753-1363	-2-10542511	03-1066051	3	24753-3463	03-1066051
20972133-2	-14774043-0	0757400-06	4	12639-3222	2472-19124
-14-0214012	10-4410949	-6-10902510	5	0	0
-14-0214012	10-4410949	-6-10902510	7	-15-1631912	0
2358-00122	5150-17702	2358-00722	11	5150-17702	0
ORBIT TYPE	ELLIPTIC	1-52791653	30	509963436	33054492-0
20-000000	1383705471	1-52791653	31	167-736052	209-16383
20-000000	179-999999	70-0362970	32	205333766	39129726-7
2999-73517	17135-9948	264-999551	33	-10-9205086	
1000-03476	24852-2631	04-990522	34	10-9205086	
	17301-2100	194-160054	35	12269-1965	
ESC = PO100 (PO100) TYPE = -0 THIS EVENT OCCURRED BY SATISFACTION OF THE CONDITION -- TIME = .120000000000E+05					

.....
 * AEROSPACE CORPORATION *
 * GENERALIZED TRAJECTORY SIMULATION SYSTEM *

INPUT DATA

```

2  JOB ORBIT ECCENTRIC ORBIT
3  BEGIN EVENT SEQUENCE
4  P010 PH10 INITIALIZE VEHICLE ON ORBIT = 3000
5  INIT INITM30 MAD = 1000
6  YACLNIM = 20
7  YANOMIN = 65
8  YANOMIN = 260
9  LYPOI = -119
10 CBOCM = 2
11 ROTATF = 0
12
13 GRAV GRAVM1
14 TECH TFOHMI
15 INTGRA INTGRM1
16
17 OUT OUTPUT
18
19 RF20 GAMMAI = 0
20
21 P0100 TIME = 12000 /STOP/
22 END EVENT SEQUENCE
23
24
25
26 END OF CASE
  
```

2.8 Remote Terminal Execution of GTS

All of the capabilities of the GTS system are available for remote terminal use in conjunction with the CDC supplied INTERCOM feature. The actual methods of operation will depend on the hardware characteristics of the terminal to be used.

The output from the job may be obtained by several different methods. The user may request the print file to be directed to his terminal where he may either list the entire file or make use of INTERCOM editing procedures to print items of particular interest. If the user has a line printer available, he may list the output by that means. Alternately, the output can be directed to the central site printer.

Figure 3 shows a portion of the input required to execute the GTS system from a remote terminal via the INTERCOM system. The INTERCOM system directs the flow of information and data between a central computer and a number of remote terminals and is compatible with CDC 6000/7000 computers. For this example the trajectory simulation that was defined in section 2.7 is executed.

The reader can easily recognize the correlation of the data given in Figure 3 and the data given in Figure 2. The additional symbols and commands are specific to the INTERCOM system and are explained in the INTERCOM User's Guide available from Control Data Corporation.

The following statements are remote terminal inputs to GTS.

```
510=JOB ORBIT    ECCENTRIC ORBIT
520=BEGIN EVENT SEQUENCE
530=POL0
540=    PH10 INITIALIZE VEHICLE ON ORBIT
550=        INIT      INITM30    RAD      = 3000
560=        HPD      = 1000
570=        INCLNIN = 20
580=        UP      = 85
590=        TANOMIN = 260
600=        LTPOL   =-119
610=        GRAV    GRAVM1    CBCON   = 2
620=        ROTATE = 0
630=        TEOM    TEOMM1
640=        INTGRA  INTGRM1    METHOD  = :AM VARIABLE STEP:
650=        ORDER  = 8
660=        DTIN   = 8
670=        OUT    OUTPUT
680=$
690=RE20 GAMMAI = 0
700=$
710=POL00 TIME = 12000 /STOP/
720=END EVENT SEQUENCE
730=$
740=$
750=END OF CASE
```

..

The following statements are INTERCOM commands used to send the above inputs to the central computer.

..save,orbit,n

..batch,orbit,input

Fig. 3. Remote Terminal Input

SECTION 3

GTS SYSTEM ARCHITECTURE

The GTS system consists of a compatible set of processors, operators, and models that can be flexibly selected to prepare a minimum software configuration customized to the requirements of a particular application. The primary functional capabilities (trajectory, vehicle sizing, optimization) can be invoked in any logical hierarchal fashion. Each of these capabilities invokes and loads only the models and operators required to accomplish its purpose. All of the models and operators are interfaced through a sophisticated data structure and corresponding set of data operators. This feature permits independently developed models to be properly interfaced, via the data structure, into a coherent simulation. Consequently, the system support structure also facilitates the extension or altering of simulations by adding, deleting, or changing model subroutines. In conjunction with GTL, the system will load and execute, in a stand-alone fashion, any individual engineering model subroutine using input defined parameters and providing corresponding outputs. This feature provides a powerful tool for the modular validation and verification of large-scale simulations.

The complete system has been automated to eliminate bothersome and time-consuming operational overhead. The required processors are invoked, in the proper sequence, on the command of CDC control cards that are automatically generated for the user by a control card generator. This allows the user to concentrate on the application and not on the physical operation of the software.

The component members of the GTS system architecture are indicated in the following paragraphs of this section along with a brief description of their function.

3.1 Control Card Generator

The control card generator constructs proper sets of CDC control cards to execute the GTS system for a variety of applications. Options are available to invoke features of a maintenance system, provide for

modifications and additions to the library of operators and models and to select, load, and execute a simulation.

3.2 GTL Processor

The GTL processor scans the GTL input data to construct files of parameter and variable data required by a given application plus determining the functional processors, associated models, and program structure required for the application. The program structure information is passed to a subsequent pre-processor that selects the required program elements from a library and constructs a tailored version of GTS. The parameter and variable information is saved on temporary files for use in the application as it is required.

3.3 GTS Pre-processor

This program module accepts files of program structure information compiled by the GTL processor. This pre-processor utilizes this information to select the required program models from the GTS system library. These models and all implicitly required supporting subroutines are structured into a computer program specific to the given application. The computer program is further processed by the CDC system loader and made available for execution. This procedure eliminates many of the problems normally associated with the ever-expanding computer memory requirements. The size and number of operators and subroutine models contained in the GTS library is allowed to grow in an open-ended fashion. This growth does not result in an ever increasing requirement for computer memory.

3.4 GTS Executives and Model Library

The primary executive operators of the GTS system were identified in Section 1.1. These functional operators (trajectory, vehicle sizing, optimization) are supported by an extensive library of numerical operators

(integration, interpolation, etc.), data operators, and physical and engineering model subroutines. Libraries are maintained in FORTRAN source language and relocatable binary format for CDC 6000/7000 series computers.

3.5 GTS Post-Processors

A series of post-processing elements are available to the user of the GTS system. Communication with these processors is via the use of disk or tape files generated during the execution of a GTS simulation. The primary post-processors provide a variety of graphical output capabilities. These graphical features provide the ability to plot any GTS variable versus any other variable. Features are also available to display information on pre-printed or computer generated world map backgrounds.